Bayesian sequential design for Copula models: a comparison of designs selected under different Copula models

Senarathne SGJ, Drovandi CC, and McGree JM
School of Mathematical Sciences, Science and Engineering Faculty,
Queensland University of Technology, Brisbane, Queensland, Australia
j.gedara@qut.edu.au

ABSTRACT

Copula models provide flexible structures to derive the joint distribution of multivariate responses. However, they are rarely considered in the experimental design context, particularly in a Bayesian framework where model and parameter uncertainty are considered. Here, we explore a variety of such models which explain dependence structures in experiments where bivariate discrete and mixed responses are observed. The sequential Monte Carlo algorithm is adopted to reduce the computational effort required in deriving efficient sequential designs. Moreover, the performance of the total entropy utility function is evaluated under different Copula models, which allows us to derive designs for the dual objectives of parameter estimation and model discrimination for Copula models.

Keywords: Model discrimination, Parameter estimation, Sequential design, Sequential Monte Carlo, Total entropy, Utility function.

1 Introduction

Bayesian experimental design provides rules to allocate resources optimally for the collection of data for experimental goals such as parameter estimation, model discrimination, and/or prediction. Research in this area is mostly restricted to experiments which yield a univariate response or a small class of multivariate responses. Copula models provide a flexible way of constructing multivariate distributions for a wide range of multiple responses. Unfortunately, such models have rarely been considered in the experimental design context due to the lack of developed methodology and approaches to overcome computational challenges in dealing with a large variety of multivariate distributions.

Copula models are multivariate distributions whose marginal distributions follow a standard uniform distribution. There are different families of Copulas, derived using Sklar's theorem (Durante and Sempi, 2010), which have been introduced in the literature (Schoelzel and Friederichs, 2008; Nelsen, 2006). Copulas are of significance to statisticians for two major reasons: firstly, as a way of examining scale-free measures of dependence; and secondly, as a starting point for constructing families of multivariate distributions. Copula models have been mostly used to describe the dependency between continuous responses, but there is a growing literature on the application of Copula models with discrete/mixed outcomes (Song et al., 2009; Tao et al., 2013).

Denman et al. (2011) provide the first approach to design experiments for bivariate binary responses described by Copula models. They have shown that designing experiments based on Copula models yield designs which lead to more precise estimates of parameter values than

the usual practice of only designing for a univariate response. However, model and parameter uncertainty were not considered. By extending the research of Denman et al. (2011), Perrone and Müller (2016) proposed an equivalence theorem for (bivariate) Copula models which allows efficient design algorithms to be formulated, and to quickly check if designs are optimal or at least efficient. However, such approaches are restricted to a small class of bivariate responses and estimation utilities. Moreover, there are currently no formal Bayesian solutions to experimental design problems which yield multivariate responses.

This paper considers deriving optimal designs for multivariate responses under a sequential Bayesian design framework. This approach is useful for designing an experiment which yields multivariate responses where a standard multivariate distribution is not available. We demonstrate our approach by deriving designs for the dual objectives of model discrimination and parameter estimation (Borth, 1975; McGree, 2017) for experiments with discrete, continuous and mixed outcomes. In this design framework, data arrive sequentially through time. As such, current prior information about the model and parameter values needs to be sequentially updated. That is, prior information are updated as new data become available such that more efficient designs can be found (Carlin et al., 1998).

To facilitate efficient Bayesian inference in sequential settings, we adopt the sequential Monte Carlo (SMC) algorithm (Del Moral et al., 2006; Drovandi et al., 2013, 2014). SMC can be seen as an extension of importance sampling (IS), and is more efficient than IS in terms of effective sample size and covering the tails of the target distribution (that is, the posterior distribution). Moreover, it has a better chance of capturing multi-modal distributions compared to Markov chain Monte Carlo (MCMC) (Tierney, 1994; Craiu and Rosenthal, 2014), and provides an efficient estimate of the model evidence (Del Moral et al., 2006) which can be used for model selection. Therefore, SMC is very useful for sequential Bayesian design (Drovandi et al., 2013, 2014). Due to the flexibility of Copula models to describe a wide range of dependence outcomes and the computational efficiencies gained through using the SMC algorithm, we propose the combination of the two can be used to locate Bayesian designs for experiments which yield multiple responses.

We highlight that this is the first study to formally undertake design to discriminate between Copula models. Further, this study is the first to undertake design for dual experimental goals of parameter estimation and model discrimination in multivariate settings. In the examples we consider in this paper, we investigate the performance of such Bayesian designs in Copula settings with the hope of providing some insight into how experiments which yield multiple responses should be designed. In particular, we explore how efficiently parameters in a Copula model can be estimated (across different Copulas), and also explore if enough information is available from realistically sized experiments to discriminate between different Copula settings. The results should provide guidance for designing future experiments which yield multivariate responses.

The outline of the paper is as follows. In Section 2, Copula models are defined, and the Copula representation of the bivariate binary/mixed outcomes is introduced. Our Bayesian sequential design framework and the total entropy utility function is defined in Section 3. To illustrate our methodology, Section 4 focuses on two examples with bivariate binary and mixed outcomes. The paper concludes with a discussion of key findings and suggestions for future research.

2 Copulas

Copula models have been extensively studied for modelling multivariate responses due to their flexibility in describing dependencies of marginal distributions (Durante and Sempi, 2010; Genest and MacKay, 1986; Nelsen, 2006). In 1959, Abe Sklar introduced the word Copula through a theorem (Durante and Sempi, 2010) describing the link between the multivariate distribution and the marginal distributions.

In a multivariate setting, Sklar's theorem states that there exists a function $C : [0,1]^d \to [0,1]$ between the multivariate cumulative distribution function (CDF) $G(y_1, y_2, ..., y_d)$ and their corresponding marginal CDFs $F_1(y_1), F_2(y_2), ..., F_d(y_d)$ such that,

$$G(y_1, y_2, ..., y_d) = C\big(F_1(y_1), F_2(y_2), ..., F_d(y_d)\big).$$

The theorem also states that, if all the marginals are continuous, then C is unique; otherwise, C is uniquely determined on the $Ran(F_1) \times Ran(F_2) \times \cdots \times Ran(F_n)$, which is the Cartesian product of the ranges of marginals.

2.1 The Fréchet-Hoeffding bounds for Copulas

Let $Y_1, Y_2, ..., Y_d$ be random variables with marginal CDFs $u_1 = F_1(y_1), u_2 = F_2(y_2), ..., u_d = F_d(y_d)$ respectively. Then, for any Copula C, there exist two boundaries W^d and M^d such that,

$$W^d(\boldsymbol{u}) \leq C(\boldsymbol{u}) \leq M^d(\boldsymbol{u}) \text{ for every } \boldsymbol{u} \text{ in } [0,1]^d \ ,$$

where $W^d(\boldsymbol{u}) = \max(u_1 + u_2 + \cdots + u_d - n + 1, 0)$ and $M^d(\boldsymbol{u}) = \min(u_1, u_2, ...u_d)$. These boundaries correspond to the perfect negative and perfect positive dependence of random variables, and referred as the Fréchet-Hoeffding lower bound and the Fréchet-Hoeffding upper bound respectively (Nelsen, 2006).

2.2 Bivariate Copulas

The bivariate Copula model is the simplest form of any Copula, which describes the dependence structure between two random variables. A large number of bivariate Copulas and their dependence properties have been discussed in the literature (Nelsen, 2006; Durante and Sempi, 2010). Several selected bivariate Copula models are described in the following subsections. Let Y_1 and Y_2 be two random variables, with distribution functions $u_1 = F_1(y_1)$ and $u_2 = F_2(y_2)$ respectively.

Product Copula

The product Copula (independence Copula) is given by

$$C(u_1, u_2) = u_1 u_2, \qquad 0 \leq u_1, u_2 \leq 1.$$

This Copula corresponds to the case of independence between the two random variables.

Frank Copula

The Frank Copula (Frank, 1979) is given by

$$C(u_1, u_2, \alpha) = -\alpha^{-1} \log\left(1 + \frac{(e^{-\alpha u_1} - 1)(e^{-\alpha u_2} - 1)}{e^{-\alpha} - 1}\right), \qquad \alpha \neq 0.$$

This Copula permits both negative and positive dependence, and it is symmetric in both tails. As the Copula parameter α approaches $-\infty, 0, \infty$, the Frank Copula approaches Fréchet-Hoeffding lower bound, independence and the Fréchet-Hoeffding upper bound, respectively.

Gumbel Copula

The Gumbel Copula (Gumbel, 1960) is given by

$$C(u_1, u_2) = \exp\Big(- \big((-\log u_1)^\alpha + (-\log u_2)^\alpha\big)^{1/\alpha}\Big), \qquad 1 \leq \alpha < \infty.$$

This Copula only permits positive dependence and exhibits strong right tail dependency. As the Copula parameter α approaches 1 and ∞, the Gumbel Copula approaches independence and the Fréchet-Hoeffding upper bound, respectively. There is no value for α such that the Copula approaches the Fréchet-Hoeffding lower bound.

Clayton Copula

The Clayton Copula (Clayton, 1978) is given by

$$C(u_1, u_2) = \big({u_1}^{-\alpha} + {u_2}^{-\alpha} - 1\big)^{-1/\alpha}, \qquad \alpha > 0.$$

This Copula only permits positive dependence and exhibits strong left tail dependency. As the Copula parameter α approaches 0 and ∞, the Clayton Copula approaches independence and the Fréchet-Hoeffding upper bound, respectively. There is no value for α such that the Copula approaches the Fréchet-Hoeffding lower bound.

Gaussian Copula

The Gaussian Copula (Joe, 1997) is given by

$$C(u_1, u_2) = \Phi_2\big(\Phi^{-1}(u_1), \Phi^{-1}(u_2); \rho\big), \qquad (u_1, u_2) \in [-1, 1]^2 \text{ and } \rho \in (-1, 1),$$

where Φ_2 is the CDF of a bivariate standard Gaussian random variable, Φ is the CDF of a standard Gaussian random variable, and ρ is the correlation coefficient between the two Gaussian random variables. This Copula is symmetric in both tails and permits negative dependence.

2.3 Copula models for multivariate data

Assume that we have data $\boldsymbol{y_{1:t}} = (\boldsymbol{y_1}, \boldsymbol{y_2}, ..., \boldsymbol{y_t})$ and design $\boldsymbol{d_{1:t}} = (d_1, d_2, ..., d_t)$ at time t, where $\boldsymbol{y_1}, \boldsymbol{y_2}, ..., \boldsymbol{y_t}$ are multivariate responses whose marginal responses can be modelled using an appropriate probability distribution function. This probability distribution may involve a link function which is used to describe the relationship between the mean of the distribution function and the predictors (design variables). For example, if $\boldsymbol{y_j} = (y_{j1}, y_{j2})$ for $j = 1, 2, ..., t$, is a bivariate binary response, both y_{j1} and y_{j2} can be modelled via a Bernoulli distribution. Then, the relationship between the mean of the distribution function and the predictors can be explained via a logit or probit link function. Subsequently, the joint distribution of y_{j1} and y_{j2} can be modelled using a suitable Copula model.

2.3.1 Copula models for bivariate binary outcomes

Suppose Y_1 and Y_2 are two binary random variables. Then, the joint probability distribution $P_{y_1,y_2} = Pr(Y_1 = y_1, Y_2 = y_2)$, $y_1, y_2 = 0, 1$ has four possible outcomes {(0,0),(0,1),(1,0),(1,1)} where '1' represents a success and '0' a failure.

The Copula representation (Denman et al., 2011) of the bivariate distribution can be expressed as

$$
\begin{aligned}
p_{11} &= C(\pi_1, \pi_2, \alpha) \\
p_{10} &= \pi_1 - p_{11} \\
p_{01} &= \pi_2 - p_{11} \\
p_{00} &= 1 - \pi_1 - \pi_2 + p_{11},
\end{aligned}
$$

where π_1, π_2 are the marginal probabilities of success and α is the Copula parameter.

Therefore, the log-likelihood for a single observation $\boldsymbol{y} = (y_1, y_2)$ is given by

$$l_i(\boldsymbol{\theta}; \boldsymbol{y}) = y_1 y_2 \log p_{11} + y_1(1 - y_2) \log p_{10} + (1 - y_1) y_2 \log p_{01} + (1 - y_1)(1 - y_2) \log p_{00},$$

where $\boldsymbol{\theta}$ is the parameter vector which involves both model parameters and the Copula parameter α.

2.3.2 Copula models for bivariate continuous and mixed outcomes

Suppose a random variable Y_1 and another random variable Y_2 are both continuous outcomes having the marginal distributions f_{Y_1} and f_{Y_2}, respectively, then the Copula representation of the joint density is given by

$$f_{Y_1,Y_2}(y_1, y_2; \alpha) = f_{Y_1}(y_1) f_{Y_2}(y_2) \frac{\partial^2 C(u_1, u_2, \alpha)}{\partial u_1 \partial u_2}.$$

If one of the random variables is discrete (say Y_2), the joint density (Tao et al., 2013) can be expressed as

$$f_{Y_1,Y_2}(y_1, y_2; \alpha) = f_{Y_1}(y_1)(C_1^* - C_2^*),$$

where $C_1^* = \frac{\partial C(u_1,u_2,\alpha)}{\partial u_1}$, $C_2^* = \frac{\partial C(u_1,u_2^-,\alpha)}{\partial u_1}$ and u_2^- is the left hand limit of u_2.

If the discrete random variable (Y_2) is a binary outcome, the joint distribution of Y_1 and Y_2 can be expressed as

$$f_{Y_1,Y_2}(y_1, y_2; \alpha) = \begin{cases} f_{Y_1}(y_1)\big(1 - C_1^*(u_1, u_2; \alpha)\big) & , y_2 = 0 \\ f_{Y_1}(y_1) C_1^*(u_1, u_2; \alpha) & , y_2 = 1 \end{cases}.$$

Therefore, the log-likelihood for a single observation is given by

$$l_i(\boldsymbol{\theta}; \boldsymbol{y}) = \log\big(f_{y_1}(y_1)\big) + (1 - y_2) \log\big(1 - C_1^*(u_1, u_2; \alpha)\big) + y_2 \log\big(C_1^*(u_1, u_2; \alpha)\big).$$

3 Design methodology

In this section, our Bayesian sequential design framework is presented. Following this, the total entropy utility function is defined in our multivariate setting along with a utility for estimation and model discrimination, and we show how these can be approximated within an SMC framework. Firstly, our Bayesian sequential design framework is described.

3.1 Bayesian sequential design

In Bayesian sequential design data arrive sequentially through time (Carlin et al., 1998). The posterior distribution expresses the uncertainty in the model and model parameters for the next data point, and hence it is necessary to update the posterior distribution as new data become available so that designs can be found based on all information currently available. This involves running a Bayesian inference algorithm to update the posterior and then maximising a utility function with respect to a given design to find the optimal design (Chaloner and Verdinelli, 1995). Although importance sampling is a commonly used technique to obtain samples from the posterior in Bayesian experimental design, it is not recommended when there is a large 'jump' between the prior and the posterior. To overcome this problem, Drovandi et al. (2013) suggested the use of SMC methods which can be applied to efficiently update the posterior distribution sequentially. SMC methods are a collection of techniques that approximate a sequence of distributions known up to a normalising constant. The approach combines importance sampling, resampling and MCMC techniques to approximate target distributions.

3.1.1 Sequential Monte Carlo Algorithm

In SMC, a set of particles representing the target distribution are traversed through a sequence of distributions connecting the prior and the posterior by repeatedly applying a set of re-weighting, resampling and move steps. In each iteration of the SMC algorithm, upon the observation of a new data point, the set of particles needs to be re-weighted such that it represents the target distribution at a given time. However, as more data are observed, the particle weights become more variable and skewed; hence the effective sample size (ESS) will decrease. When the ESS drops below a threshold, particles are resampled and moved. The resampling technique is used to increase the ESS back up to the initial sample size. After the resampling step, the current particle set will mostly likely contain many duplicate particles, and hence the move step is required to diversify the set.

When model uncertainty is taken into account, the SMC algorithm can be run in parallel for each model. That is, the inference for each model is undertaken separately, and the models are only considered jointly when evaluating the utility of a given design.

Consider a new design point d_{t+1}, where $\boldsymbol{y_{t+1}}$ is the next observed outcome. At the current target at time t, we have a particle set $\{W^i_{m,t}, \boldsymbol{\theta^i_{m,t}}\}^N_{i=1}$ for $m = 1, 2, ..., K$, where $W^i_{m,t}$ are the particle weights for the i^{th} particle of model m at time t. Upon the observation of $\boldsymbol{y_{t+1}}$ from design d_{t+1}, the current particle set is reweighted for each model as follows

$$w^i_{m,t+1} = W^i_{m,t} f(\boldsymbol{y_{t+1}}|m, \boldsymbol{\theta^i_{m,t}}, d_{t+1}) \text{ for } i = 1, 2, ..., N,$$

where $w^i_{m,t+1}$ are the unnormalized importance weights for the data $(d_{t+1}, \boldsymbol{y_{t+1}})$. Once the weights are normalised to yield $W^i_{m,t+1}$, the particle set $\{W^i_{m,t+1}, \boldsymbol{\theta^i_{m,t}}\}^N_{i=1}$ approximates the appropriate target distribution. If the effective sample size ESS_m of the particle set $\{W^i_{m,t+1}, \boldsymbol{\theta^i_{m,t}}\}^N_{i=1}$

is less than a threshold E, multinomial sampling is used in the resample step to duplicate particles with high weight and remove particles with relatively low weight which gives $\{1/N, \boldsymbol{\theta}^{\boldsymbol{i}}_{\boldsymbol{m,t}}\}_{i=1}^{N}$. Then, in the move step, the particle set is diversified via an MCMC kernel with an invariant distribution $\pi_{t+1}(\boldsymbol{\theta}^{\boldsymbol{i}}_{\boldsymbol{m,t+1}}|m, \boldsymbol{y_{1:t+1}}, \boldsymbol{d_{1:t+1}})$, where $\boldsymbol{\theta}^{\boldsymbol{i}}_{\boldsymbol{m,t+1}}$ is the i^{th} particle of m^{th} model at time $t+1$, and $\boldsymbol{y_{1:t+1}}$ and $\boldsymbol{d_{1:t+1}}$ are data and design points collected up to time $t+1$, respectively. Here, a random walk Metropolis-Hastings algorithm is used to propose new particles R times. We note that the random walk variance can be chosen based on the variance-covariance of the current particle set as it already approximates the current target distribution. Hence, efficient choices for the tuning parameters in this Metropolis-Hastings algorithm can be made throughout the experiment without any user intervention.

The predictive distribution of $\boldsymbol{y_{t+1}}$ is the distribution of the unobserved data $\boldsymbol{y_{t+1}}$ conditional on the observed data $\boldsymbol{y_{1:t}}$ and design points $\boldsymbol{d_{1:t}}$. It is often used in finding Bayesian designs, particularly when searching for the next optimal design under parameter and model uncertainty, and is given by:

$$\int_{\boldsymbol{\theta}} f(\boldsymbol{y_{t+1}}|m, \boldsymbol{\theta}, \boldsymbol{d_{1:t+1}}) p(\boldsymbol{\theta}|m, \boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) \mathrm{d}\boldsymbol{\theta}.$$

As shown by Del Moral et al. (2006), this is equivalent to $Z_{m,t+1}/Z_{m,t}$, the ratio of normalising constants of successive target distributions. Thus, when implementing SMC, we can approximate this ratio as follows

$$Z_{m,t+1}/Z_{m,t} \approx \sum_{i=1}^{N} w^{i}_{m,t+1}, \tag{1}$$

and the (log) model evidence can be approximated as follows

$$\log \hat{Z}_{m,t+1} = \sum_{i=0}^{t} \log \left(\hat{Z}_{m,t+1-i} / \hat{Z}_{m,t-i} \right).$$

Thus, the model evidence $Z_{m,t}$ is updated each time a new data point is observed (noting that $Z_{m,0} = 1$). After approximating the model evidence of each model, they can be normalized to estimate the posterior model probability, $\hat{\pi}(m|\boldsymbol{y_{1:t+1}}, \boldsymbol{d_{1:t+1}})$ for $m = 1, 2, ..., K$ as follows

$$\hat{\pi}(m|\boldsymbol{y_{1:t+1}}, \boldsymbol{d_{1:t+1}}) = \frac{\hat{Z}_{m,t+1}}{\sum_{m=1}^{K} \hat{Z}_{m,t+1}}. \tag{2}$$

3.1.2 Bayesian sequential design for multivariate setting

The SMC algorithm has been developed in Bayesian design for model discrimination (Drovandi et al., 2014), parameter estimation (Drovandi et al., 2013; McGree et al., 2016) and the dual purpose of model discrimination and parameter estimation (McGree, 2017). We adopt this algorithm to derive optimal designs such that they can be applied to Copula models. The SMC algorithm is as follows (Drovandi et al., 2014).

Algorithm 1 SMC

1: Draw $\boldsymbol{\theta}^{\boldsymbol{i}}_{\boldsymbol{m,0}}$ from prior $p(\boldsymbol{\theta_m}|m)$ and set $W^i_{m,0} = 1/N$ for $i = 1, 2, ..., N$ and for $m = 1, 2, ..., K$
2: Set $\log \hat{Z}_{m,0} = 0$ for $m = 1, 2, ..., K$
3: **for** $t = 0$ to $T - 1$ **do**
4: Find the $(t + 1)^{th}$ optimal design point, d_{t+1}, by maximising the utility $U(d_{t+1}|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}})$
5: Collect data $\boldsymbol{y_{t+1}}$ at the design point d_{t+1} (In simulation studies, data are generated from an assumed true model)
6: **for** $m = 1$ to K **do**
7: Compute the likelihood $l^i_{m,t+1}(\boldsymbol{\theta^i_{m,t}}, \boldsymbol{y_{t+1}}) = f(\boldsymbol{y_{t+1}}|\boldsymbol{m}, \boldsymbol{\theta^i_{m,t}}, \boldsymbol{d_{1:t+1}})$ using a Copula model
8: Reweight $w^i_{m,t+1} = W^i_{m,t}\ l^i_{m,t+1}(\boldsymbol{\theta^i_{m,t}}, \boldsymbol{y_{t+1}})$ for $i = 1, 2, ..., N$
9: Update log evidence $\log \hat{Z}_{m,t+1} = \log \hat{Z}_{m,t} + \log \sum_{j=1}^N w^j_{m,t+1}$
10: Normalize the weights $W^i_{m,t+1} = w^i_{m,t+1} / \sum_{j=1}^N w^j_{m,t+1}$ for $i = 1, 2, ..., N$
11: Calculate $\text{ESS}_m = 1/\sum_{i=1}^N (W^i_{m,t+1})^2$
12: **if** $\textbf{ESS}_m < E$ **then**
13: Resample particle set m producing $\{\boldsymbol{\theta^i_{m,t+1}}, W^i_{m,t+1}\}_{i=1}^N$
14: Compute the parameters of the MCMC proposal $q_{m,t+1}(.|.)$ using the particles $\{\boldsymbol{\theta^i_{m,t+1}}, W^i_{m,t+1}\}_{i=1}^N$
15: **for** $i = 1$ to N **do**
16: Move particle $\boldsymbol{\theta^i_{m,t+1}}$ with an MCMC kernel of invariant distribution $\pi_{t+1}(\boldsymbol{\theta^i_{m,t+1}}|\boldsymbol{y_{1:t+1}}, \boldsymbol{d_{1:t+1}})$ iterated R times
17: **end for**
18: Reset $W^i_{m,t+1} = 1/N$ for $i = 1, 2, ..., N$
19: **else**
20: Set $\boldsymbol{\theta^i_{m,t+1}} = \boldsymbol{\theta^i_{m,t}}$
21: **end if**
22: **end for**
23: **end for**

In the first step of the SMC algorithm, a sample of N particles $\{\boldsymbol{\theta}^i_{\mathbf{m,0}}, W^i_{m,0}\}_{i=1}^N$ is drawn from the prior for each model m. Then, at each time step, a Bayesian design is found by maximising a utility function denoted by $U(d|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}})$ based on the current information about each model and the parameters (line 4). Given this design, a new data point is observed. In this desktop study, data cannot actually be observed. In place of this, data are simulated from a supposed true Copula model (line 5). Based on this new data point, the weights of the particle set and model evidence for each model are updated (lines 7 to 10), and the ESS_m is approximated for each model (line 11). If the ESS_m is less than a predefined threshold E, then the resample weights of the particle set and model evidence for each model are updated and move steps are undertaken (lines 13 to 16). This process continues until a fixed number of data points have been observed.

3.2 Utility function

The utility function for a proposed design point, d, given current data and design $(\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}})$ can be expressed as $U(d|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}})$, where d is selected from a finite set of design points, $\mathcal{D}$. In the Bayesian context, the utility function can be expressed as the expectation of a function of

the posterior distribution of unknowns as follows

$$U(d|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) = \sum_{m=1}^{K} \pi(m|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) \int_{\mathcal{S}} \int_{\boldsymbol{\theta_m}} U(d, \boldsymbol{z}, m|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) f(\boldsymbol{z}|\boldsymbol{\theta_m}, m, \boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}, d) p(\boldsymbol{\theta_m}|m) d\boldsymbol{\theta_m} d\boldsymbol{z}, \tag{3}$$

where $\boldsymbol{z}$ is a potential outcome for the proposed design point d. The outcome $\boldsymbol{z}$ belongs to the sample space $\mathcal{S}$. For example, in the bivariate binary case, $\mathcal{S} = \{(0,0), (0,1), (1,0), (1,1)\}$, and the relevant integral above would be a sum over these four possible outcomes.

When continuous or mixed data are observed, Monte Carlo (MC) integration can be applied to estimate the above utility as follows

$$\hat{U}(d|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) = \sum_{m=1}^{K} \hat{\pi}(m|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) \frac{1}{q} \sum_{j=1}^{q} U(d, \boldsymbol{z_j}, m|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}),$$

where $\boldsymbol{z_j} \sim f(\boldsymbol{z}|\boldsymbol{\theta_m^j}, m, \boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}, d)$ and $\boldsymbol{\theta_m^j} \sim p(\boldsymbol{\theta_m}|m, \boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}})$. Here, $\left\{(\boldsymbol{z_j}, \boldsymbol{\theta_m^j}), j = 1, 2, ..., q\right\}$ is a MC sample.

When discrete data are observed, the utility (Eq.(3)) can be estimated as follows

$$\hat{U}(d|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) = \sum_{m=1}^{K} \hat{\pi}(m|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) \sum_{\boldsymbol{z} \in \mathcal{S}} f(\boldsymbol{z}|m, \boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}, d) U(d, \boldsymbol{z}, m|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}).$$

Next we define the total entropy utility function. However, to do this, we must first define utilities for estimation and discrimination.

3.2.1 Parameter estimation utility

The Kullback Leibler distance (KLD; Kullback and Leibler (1951)) is a utility function that can be used for parameter estimation in Bayesian design. The utility $U_P(d, \boldsymbol{z}, m|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}})$ is the KLD between the prior and posterior distribution based on potentially observing $\boldsymbol{z}$ from design d. This can be expressed as follows

$$\int_{\boldsymbol{\theta_m}} p(\boldsymbol{\theta_m}|m, \boldsymbol{z}, \boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}, d) \log\left(f(z|\boldsymbol{\theta_m}, m, \boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}, d)\right) \mathrm{d}\boldsymbol{\theta_m} - \log\left(\frac{f(\boldsymbol{y_{1:t}}, \boldsymbol{z}, \boldsymbol{m}|\boldsymbol{d_{1:t}}, d)}{f(\boldsymbol{y_{1:t}}, \boldsymbol{m}|\boldsymbol{d_{1:t}})}\right).$$

Using the approximation in Eq.(1) and a particle approximation to the above integral, KLD can be approximated in sequential design as follows (Drovandi et al., 2013),

$$\hat{U}_P(d, \boldsymbol{z}, m|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) = \sum_{i=1}^{N} W_{m,t}^{i} \log\left(f(\boldsymbol{z}|\boldsymbol{\theta_{m,t}^i}, m, d)\right) - \log \sum_{i=1}^{N} w_{m,t}^{i}. \tag{4}$$

3.2.2 Model discrimination utility

Here, we discuss the mutual information utility function for model discrimination based on the mutual information between the model indicator, m, and the observation, $\boldsymbol{z}$, given data $\boldsymbol{y_{1:t}}$ and design $\boldsymbol{d_{1:t}}$ (Drovandi et al., 2014).

The mutual information utility for model discrimination for a proposed design point, d, can be expressed as follows

$$U_M(d, \boldsymbol{z}, m|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) = \log \pi(m|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}, \boldsymbol{z}, d). \tag{5}$$

In our SMC algorithm (see Section 3.1) we have discussed how to approximate the posterior model probabilities as new data become available. By substituting those approximations into Eq.(5), we can approximate the utility of a design point d upon the observation of z for a given model m.

3.2.3 Total entropy utility

Total entropy measures both the uncertainty about the model and about the parameter values. By following the work of Borth (1975), we define total entropy as follows

$$U_T(d|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) = U_M(d|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}) + U_P(d|\boldsymbol{y_{1:t}}, \boldsymbol{d_{1:t}}), \tag{6}$$

and this can be applied when discrete data or otherwise are observed. Using the approximations as given in Equations (2) and (4), the total entropy utility can be approximated.

4 Applications

The following two examples apply and investigate the performance of Algorithm 1 and total entropy in deriving optimal designs for the dual purpose of model discrimination and parameter estimation under various Copula models. First, we consider an example from Denman et al. (2011) where bivariate binary outcomes are observed. Then, we consider an example from pharmacology (Tao, 2010) where bivariate mixed outcomes are observed.

For optimal design selection, the total entropy utility (Eq.(6)) was implemented within the SMC algorithm. We benchmark the performance of the total entropy utility against random selection. It should be noted that a comparison between total entropy and both KLD and mutual information for model discrimination utilities was also undertaken. Here, similar results to McGree (2017) were found. That is, total entropy performs well under both utilities with little compromise between the two. As such, those results are omitted, and we instead investigate the performance of total entropy under different Copula models. To do this, a simulation study was undertaken where experiments were sequentially simulated within our design framework. Within this framework, the next data point is generated based on the selected optimal design and an assumed true Copula model with particular parameter values. The parameter vector of the true Copula model contains the model parameters and the dependence parameter. Once the data have been generated, they will used to update the prior information. The SMC algorithm will run until a fixed number of data points have been observed.

Throughout the examples, we use $N = 5000$ particles and a re-sampling threshold of 75% ($E = 3750$). Both examples have a discrete design space, and hence the Brute Force Grid Search (BFGS) algorithm was used to select the next optimal design point d_{t+1}. In the SMC algorithm, particles were moved through the MCMC kernel $R = 30$ times. As the results are subject to variability through the simulated data, 500 simulations were used for the each simulation study.

4.1 Example 1

Following the work of Denman et al. (2011), consider an example with two binary responses which were modelled by a three-factor main effects logistic GLM with $x_j \in \{-1, 0, 1\}$, for $j = 1, 2, 3$ as follows

$$\log\left(\frac{\pi_1}{1-\pi_1}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3, \tag{7}$$

$$\log\left(\frac{\pi_2}{1-\pi_2}\right) = \gamma_0 + \gamma_1 x_1 + \gamma_2 x_2 + \gamma_3 x_3. \tag{8}$$

For data generation, parameter values were taken as $\boldsymbol{\beta} = [1, 4, 1, -1]$ and $\boldsymbol{\gamma} = [1, -.5, 1, -1]$, see Denman et al. (2011). Denman et al. (2011) considered a locally optimal design approach to locate designs for parameter estimation. Here, we consider this example under a fully Bayesian design framework with model uncertainty, and consider total entropy as the utility function for design selection.

The prior distributions of the model parameters were independent normal distributions with mean 0 and variance 16, and the prior distribution for the Copula parameter was a mixture of uniform distributions, $(U[-18, -0.5] \cup U[0.5, 18])$. Two design scenarios were considered which differ in terms of the discrimination problem. Firstly, we consider the problem of discriminating between different Copula functions, and secondly we discriminate between different linear predictors. In each case, two candidate models are considered, and each model is equally likely a priori.

4.1.1 Design scenario 1

In the first scenario, the above two logistic regression models (Eq.(7) and Eq.(8)) were used to model the marginal responses, and the Copula models were created via the Frank and product Copula. For the Frank Copula, it was assumed that $\alpha = 10$ for data generation.

Results: Fig.1 compares the distribution of the selected optimal designs when each Copula model was responsible for data generation. This figure shows that for both Copula models, covariates x_2 and x_3 preferred either -1 or +1 while the covariate x_1 preferred the value 0. In particular, the design points $(0, -1, .)$ and $(0, 1, 1)$ were frequently selected by the total entropy utility, and those selected designs were similar to the designs obtained from the estimation utility. In order to compare the performance of the total entropy utility for model discrimination, the posterior model probabilities for the true model were plotted against time (see Fig.2). From these results, it can be seen that the posterior model probabilities of optimal designs converge to one in a fewer number of iterations than the posterior model probabilities of the random design. The results indicate that it is possible to discriminate between different Copula functions with it being comparatively easier to determine there is dependence in the responses then it is to determine that they are independent. Indeed, when the Frank Copula is generating data, only about 50 multivariate observations are needed to determine that it is the true model while around 200 multivariate observations are needed to determine the product Copula was generating data.

Then, we assessed posterior precision of the parameters when designs were found via total entropy and random selection. For the comparison, we evaluated the log determinant of the variance-covariance matrix of the posterior distribution. Fig.3 shows the boxplots of the log determinant values of each intermediate posterior distribution (for each design point) for all 500 simulations. In Fig.3, both plots (a) and (c) have lower log determinant values compared

to the plots (b) and (d), which means the posterior distributions obtained from the optimal design have higher precision compared to those obtained from the random design. In terms of comparing Copula functions, the total entropy utility function appears to estimate parameters in both functions well, with similar performance when each function was generating data.

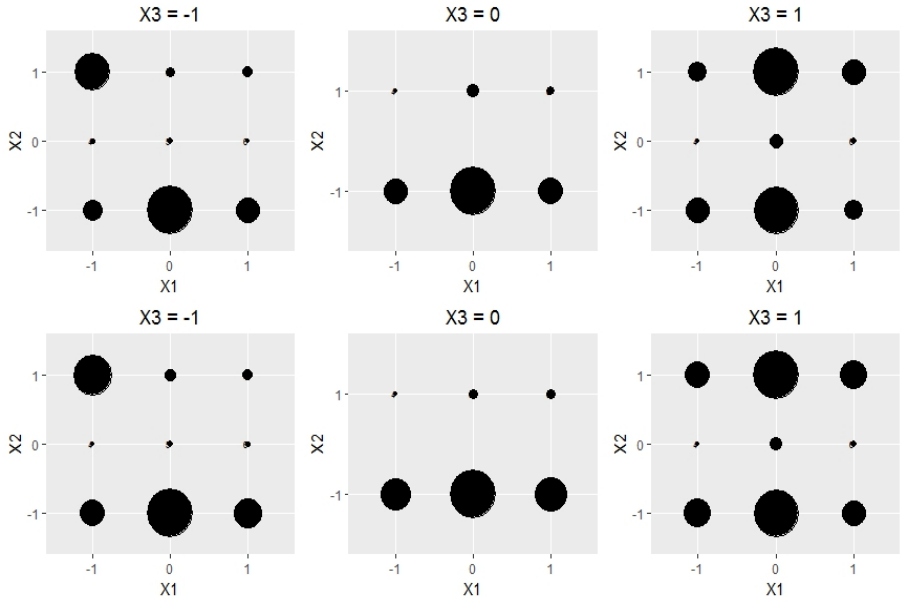

Figure 1: The selected optimal design points over 500 simulations when the Frank Copula model is true (first row) and when the product Copula model is true (second row) for scenario 1.

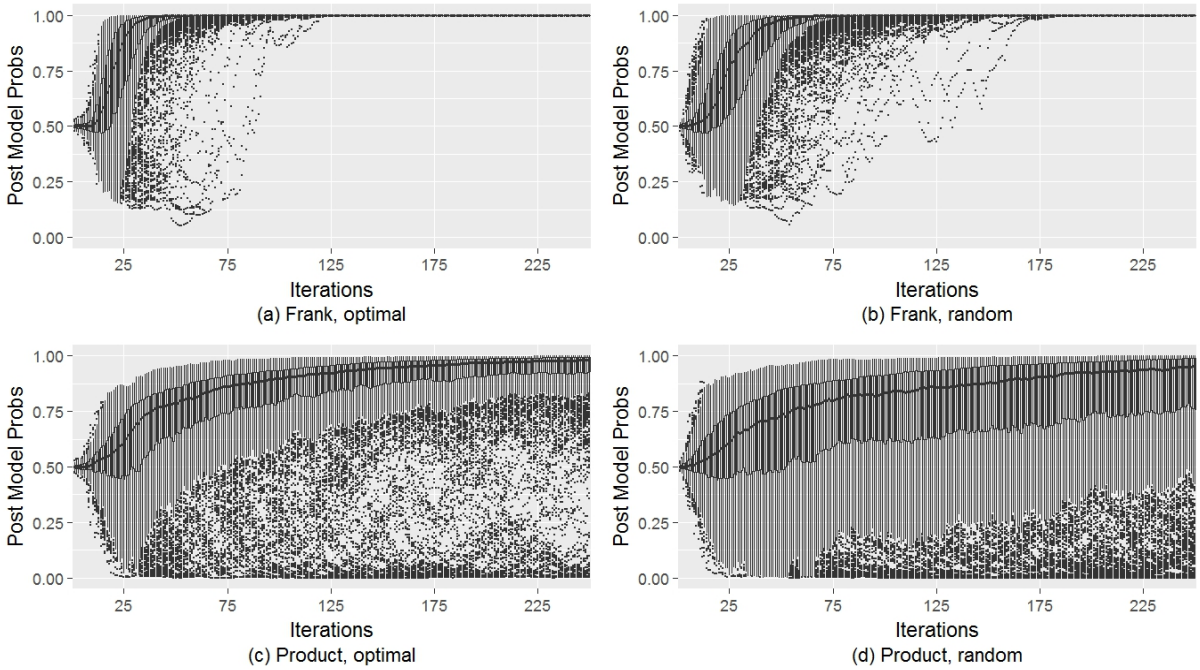

Figure 2: The boxplots of the posterior model probabilities of true model over 500 runs for 250 subjects for scenario 1. In (a) and (b), the Frank Copula model is true, where (a) is the optimal design and (b) is the random design. In (c) and (d), the product Copula model is true, where (c) is the optimal design and (d) is the random design.

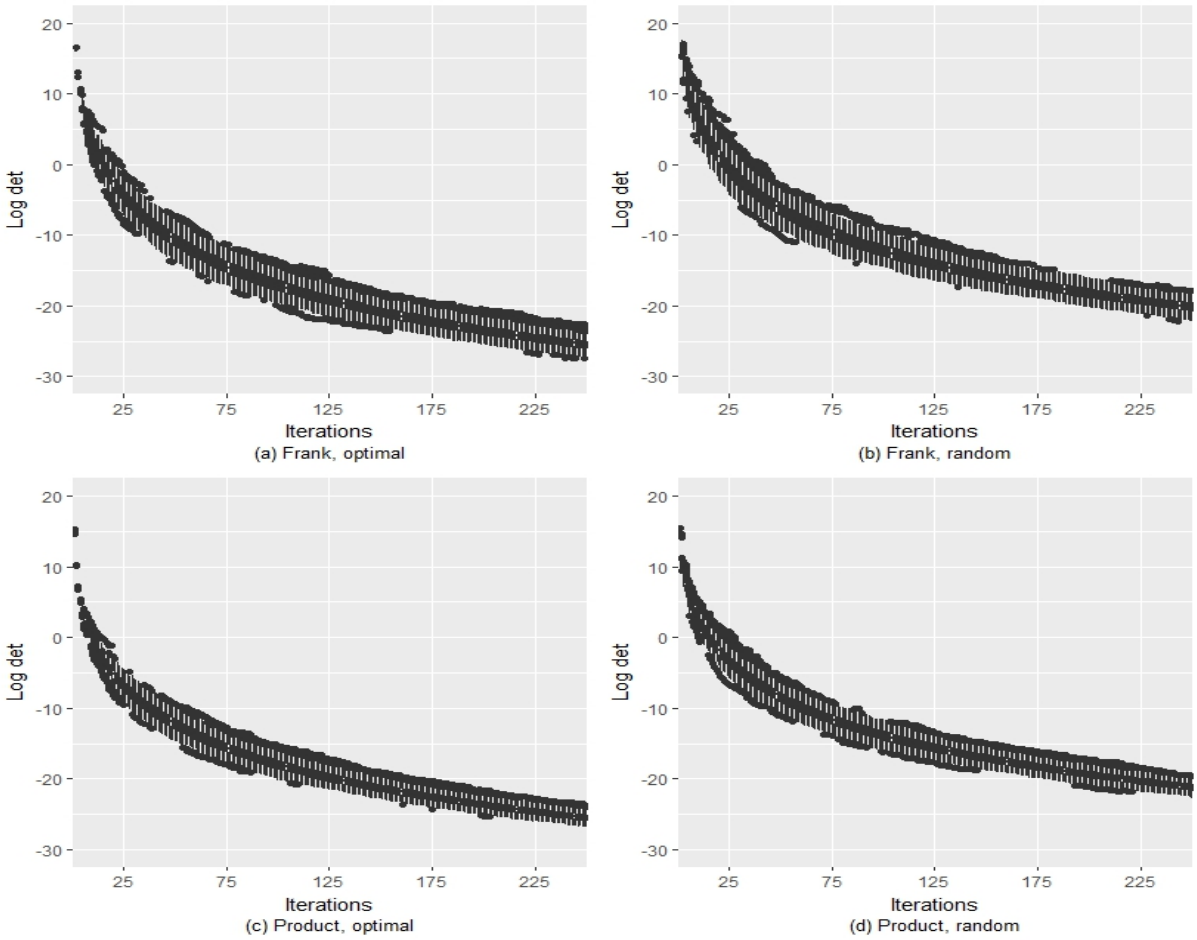

Figure 3: The boxplots of the logarithm of the determinant of the posterior variance-covariance matrix for each design point over 500 simulations for 250 subjects for design scenario 1. In (a) and (b), the Frank Copula model is true, where (a) is the optimal design and (b) is the random design. In (c) and (d), the product Copula model is true, where (c) is the optimal design and (d) is the random design.

4.1.2 Design scenario 2

In the second scenario, the same binary response models with different covariates were considered with the intention of discriminating between linear predictors. In the first Copula model, both binary responses (y_1, y_2) were modelled by using three covariates (x_1, x_2, x_3) as shown in Equations (7) and (8). However, in the second Copula model, only two covariates (x_1, x_2) were used to model the same two binary responses as follows

$$\log\left(\frac{\pi_1^*}{1-\pi_1^*}\right) = \beta_0 + \beta_1 x_1 + \beta_2 x_2,$$

$$\log\left(\frac{\pi_2^*}{1-\pi_2^*}\right) = \gamma_0 + \gamma_1 x_1 + \gamma_2 x_2.$$

It was assumed that there is a strong positive dependence between the two binary responses. Therefore, the Frank Copula model ($\alpha = 10$) was used to induce dependence between the two responses. All parameter values considered when generating data were as described in the first scenario (as appropriate). Similarly, the same prior distributions were assumed.

Results: Fig.4 shows the distribution of selected optimal designs throughout the simulation study when each Copula model was generating data. According to Fig.4, the selected designs were slightly different for two Copula models. However, similar to the first scenario, most of the optimal design points had either -1 or +1 for the covariates x_2 and x_3, and 0 for the covariate x_1. The preference for x_3 to be placed on the boundary of the design space seems sensible as the responses would then be measured at the extremes of this covariate. Comparing responses at such extremes should provide information about whether this covariate is needed in the model. Also, the selected dual purpose designs were similar to the designs selected via the estimation utility suggesting that the largest reduction in entropy occurred by focusing on estimation.

In Fig.5, the posterior model probabilities of the true Copula model over each iteration of the SMC algorithm were plotted to compare the model discrimination results of optimal and random designs for different Copula models. Here, it can be seen that both optimal and random designs perform well for discriminating between linear predictors. That is, it appears to be relatively straightforward to discriminate between different linear predictors when the Frank Copula is considered, and it appears to require a similar number of observations to determine whether a covariate is needed or can be removed.

For the purpose of comparing the parameter estimation results, log determinant values of the posterior variance-covariance matrix were evaluated and plotted. Fig.6 shows that the posterior distributions based on the optimal designs have smaller log determinant values compared to those based on random designs. Further, neither Copula model seems to be more difficult to estimate parameter values with similar precisions observed under each model.

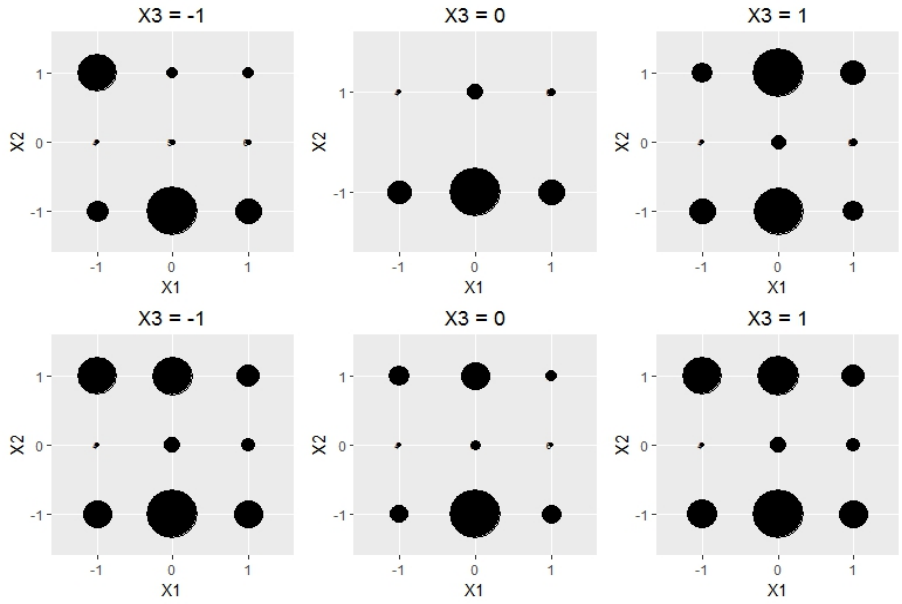

Figure 4: Optimal design points when the first Copula model is true (first row) and when the second Copula model is true (second row) for scenario 2.

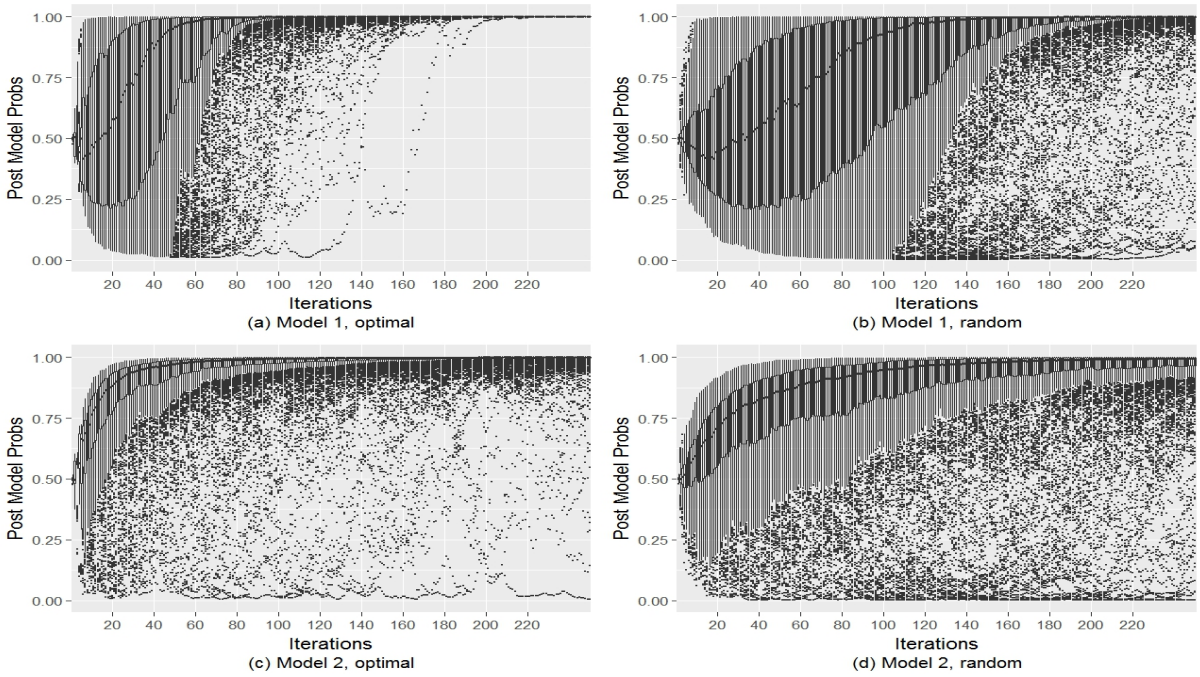

Figure 5: The boxplots of the posterior model probabilities of true model over 500 runs for 250 subjects for scenario 2. In (a) and (b), the first Copula model is true, where (a) is the optimal design and (b) is the random design. In (c) and (d), the second Copula model is true, where (c) is the optimal design and (d) is the random design.

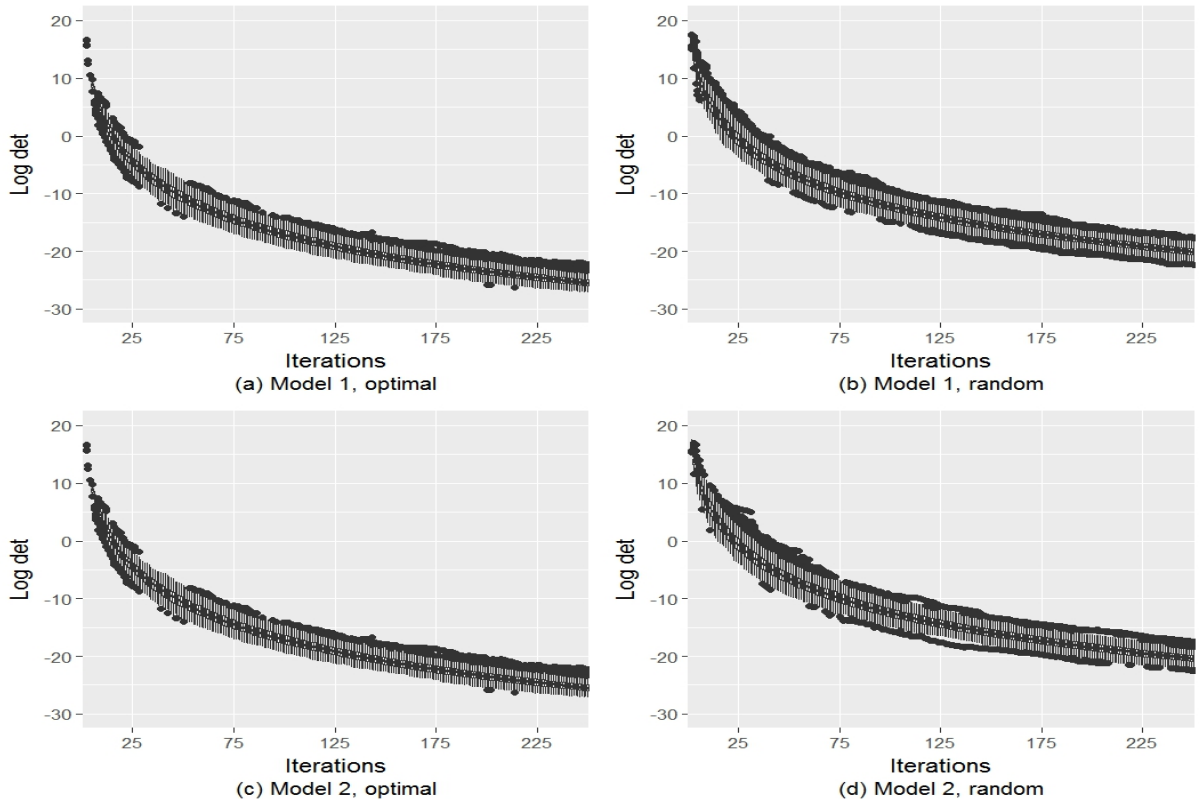

Figure 6: The boxplots of the logarithm of the determinant of the posterior variance-covariance matrix for each design point over the 500 simulations for scenario 2. In (a) and (b), the first Copula model is true, where (a) is the optimal design and (b) is the random design. In (c) and (d), the second Copula model is true, where (c) is the optimal design and (d) is the random design.

4.2 Example 2

Motivated by the work of Tao (2010), we assess the performance of Algorithm 1 in finding optimal doses for a clinical trial of Angiotensin-converting enzyme (ACE) inhibitors for prevention of hypertension and heart failure. In their work, a Copula model was fitted to combine the continuous efficacy outcome with the binary toxicity outcome. The efficacy outcome is the change of diastolic blood pressure from baseline which follows a normal distribution based on an E-max model as follows

$$Y_{1i} \sim N(f(d_i, \boldsymbol{\beta}), \sigma^2), \quad f(d_i, \boldsymbol{\beta}) = \beta_0 + \beta_{max} d_i/(\beta_{50} + d_i),$$

where β_0 represents the basal effect (corresponding to the responses when the dose is zero), β_{max} is the maximum effect attributable to the drug, and β_{50} is the dose which produce half of the β_{max} effect.

The toxicity outcome measures whether the glomerular filtration rate (GFR) decrease from baseline is greater than a threshold value. If so, then this is considered a success otherwise it is a failure (Tao, 2010). This measure of toxicity was assumed to follow a Bernoulli distribution based on a logistic regression model as follows

$$Y_{2i} \sim Ber(g(d_i, \boldsymbol{\gamma})), \quad g(d_i, \boldsymbol{\gamma}) = 1/(1 + \exp(-\gamma_0 - \gamma_1 d_i)),$$

where γ_0 and γ_1 are the intercept and the slope of the logistic regression model, respectively.

For data generation, it was assumed that $\boldsymbol{\beta} = [2.5, 14.5, 0.2]$, $\boldsymbol{\gamma} = [-2.123, 3.728]$ and $\sigma = 0.4$, see Tao (2010). This study was undertaken in a restricted design space with only a fixed number of doses being available. That is,

$$d = \{0.05n \mid n \in (1, 2, ...40)\}.$$

In this example, five Copula models (Product, Frank, Gumble, Clayton, and Gaussian Copula, see Section 2.1) were considered when generating data in Algorithm 1 (line 5). We assumed that there is a strong association between the two variables, and hence all Copula parameters (except for the Gaussian Copula) were set to 20. We set the Gaussian Copula parameter to 0.9 to impose a similarly strong positive association between the two outcomes. Initially, each Copula model had an equal chance of being selected, and therefore the prior model probability of each Copula model was taken as 0.2.

The prior distributions of the model parameters were normal with the prior means being approximately equal to the true model parameters, and the Copula parameters were selected from the uniform distribution by assuming the correct sign (positive) of the association between the two responses. Details given in Table 1.

Table 1: Prior distributions of the model parameters and the Copula parameters

Model parameter	Prior distribution	Copula parameter	Prior distribution
$\log(\beta_0)$	$N\big(\log(2), 0.5^2\big)$	Frank	$U[1, 30]$
$\log(\beta_{max})$	$N\big(\log(13), 0.5^2\big)$	Gumbel	$U[1, 30]$
$\log(\beta_{50})$	$N\big(\log(0.25), 0.5^2\big)$	Clayton	$U[1, 30]$
γ_0	$N\big(0, 3^2\big)$	Gaussian	$U[0, 0.99]$
γ_1	$N\big(0, 3^2\big)$		

In each case, a particular Copula model was assumed as the true model, and hence was responsible for generating data in Algorithm 1, line 5. Then, the posterior model probabilities of each Copula model and the posterior precision under the optimal design and that of the random design were compared in terms of being able to discriminate between models and estimate parameters.

Results: Fig.7 shows the distribution of the derived optimal doses when each Copula model was responsible for data generation. As can be seen, the distribution of selected optimal doses is similar for all Copula models despite the fact that the dependency structure of one Copula can be significantly different from another Copula. However, for the Gaussian Copula, a large number of doses were selected from dose levels which are comparatively higher than that of all other Copula models. For all Copula models, the distributions of the selected doses were similar to that of when doses were selected via the estimation utility. However, when the discrimination utility was employed for the selection of doses, both Gaussian and Gumbel Copula preferred higher dose levels while Clayton Copula preferred lower dose levels. Further, it was noted that the distribution of the doses selected from the discrimination utility had a unique shape for each Copula model. After deriving these optimal designs, we assessed the model discrimination and parameter estimation results of each Copula model separately.

Fig.8 shows the posterior model probability of the true Copula model over each iteration of the SMC algorithm, which is used to compare the model discrimination results of optimal and random designs for different Copula models. The results show that using total entropy here for design selection only marginally improves the ability to discrimination between Copula models over using random selection. This may be because of the restricted design space but McGree (2017) found similar results for one dimensional design spaces. That is, the random design was relatively efficient for discrimination. In terms of comparing Copula functions, it seems relatively straightforward to determine the Gumbel and product Copulas were responsible for data generation. It appears to be more difficult to determine when the Frank and Clayton Copulas were generating data with the posterior model probabilities relatively slowly approaching 1. The most challenging discrimination problem occurred when the Gaussian Copula was generating data. Here, even after 225 multivariate observations, we were unable to say with high certainty that this is the preferred Copula. Upon further investigation, it seems difficult to discriminate between the Frank and Gaussian Copula (particularly when the Gaussian Copula is true). This may be reasonable as the two Copula functions induce similar forms of dependence between the two responses.

Fig.9 compares the parameter estimation results of each Copula model when using optimal and random designs for different Copula models. According to Fig.9, for all selected Copula models, the posterior distributions produced by the optimal doses have lower log determinant values compared to that of random designs. The total entropy utility function appears to estimate parameters equally well when compared to the random design for all Copula models.

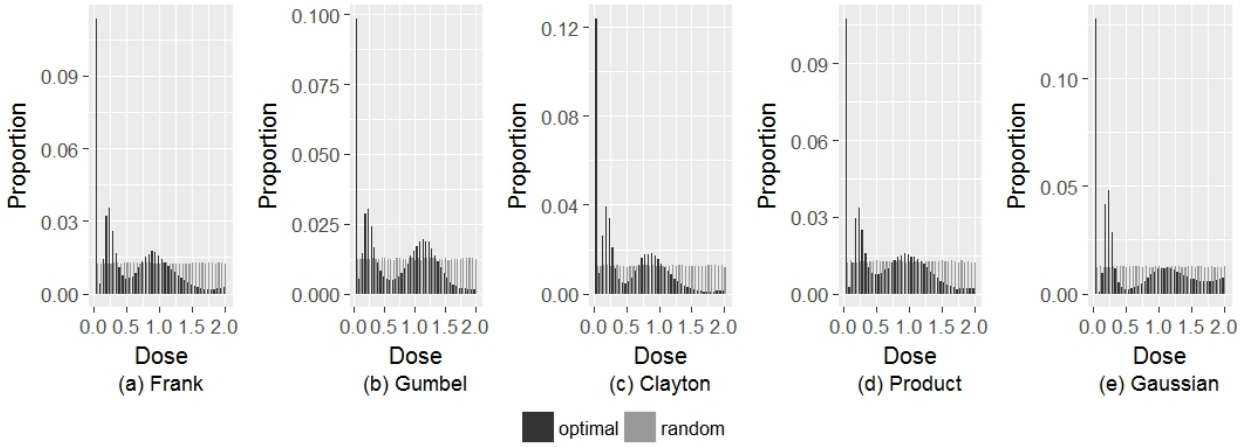

Figure 7: The distributions of the selected designs under each Copula model.

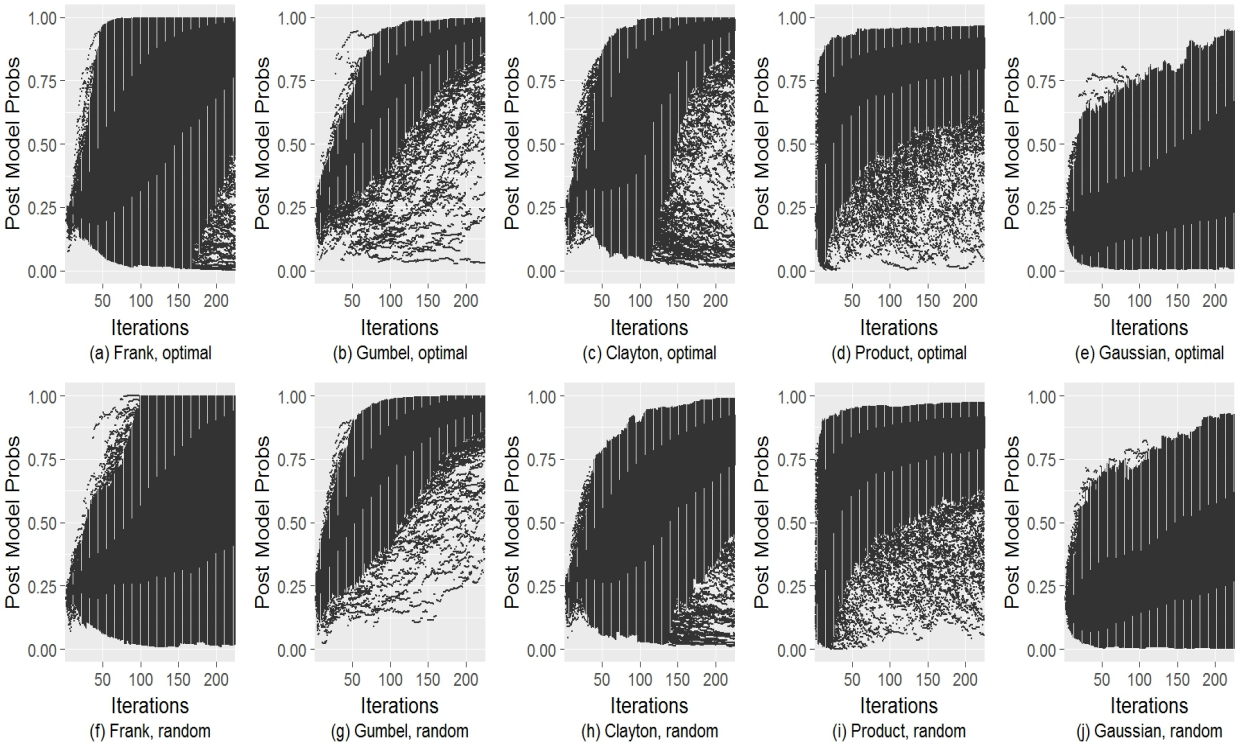

Figure 8: The distribution of the posterior model probabilities of the true Copula model, with optimal designs (row 1) and random designs (row 2), over 500 runs for 225 subjects.

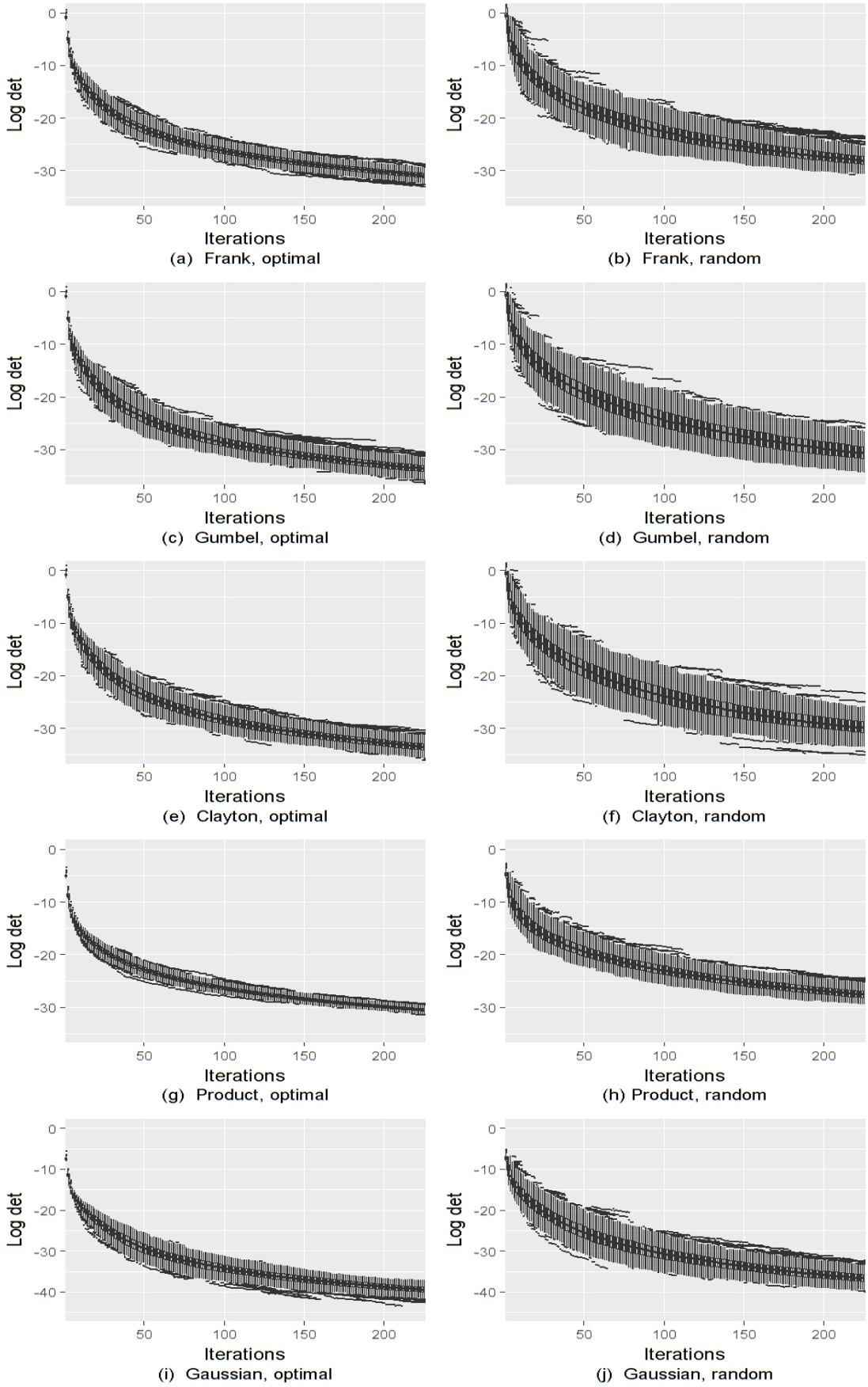

Figure 9: The boxplots of the logarithm of the determinant of the posterior variance-covariance matrix for each design point over the 500 simulations.

5 Discussion

In this work, we have developed a Bayesian sequential design approach to derive optimal designs for experiments with multivariate responses described by Copula models. The derived optimal designs can be used for the dual objectives of parameter estimation and model discrimination of Copula models. The empirical evidence presented in this paper demonstrates that the identified Bayesian sequential framework can be used to discriminate between different linear predictors and most dependency structures of bivariate responses described by Copula models and to efficiently estimate model parameters.

A GLM example with bivariate binary data and an application in pharmacology with mixed outcomes were considered to demonstrate the performance of our design approach and the total entropy utility in designing experiments for bivariate responses described by Copula models. In this first example, two design scenarios were considered where the first scenario was used to discriminate between different Copula models while the second scenario was used to discriminate between different linear predictors. The model discrimination results for these two scenarios confirmed that it is possible to discriminate between different Copula models with dependent (combined via the Frank Copula) and independent responses. Also, it appears to be relatively straightforward to discriminate between different linear predictors when the Frank Copula is considered. Furthermore, the total entropy utility function appears to estimate parameters equally well for both Copula models considered in this example.

The dose finding example (Example 2) revealed that the optimal design did not show a significant gain over random design in terms of discriminating between Copula models, but this appears to be typical for one-dimensional design settings. Further, in cases where Copulas induce similar dependence between the responses, it can be challenging to determine which model is preferred. This was observed in Example 2 when the Frank and Gaussian Copulas were considered. Both Copulas have a symmetric dependence structure, and this may be why it was difficult to discriminate between the two. In such cases, it may be more appropriate to only consider one Copula function to describe a particular form of dependence rather than considering multiple Copulas with only subtle differences in the dependence structure. The parameter estimation results of Example 2 again revealed that it is possible to efficiently estimate model parameters across many different Copula models.

Most of the existing experimental design approaches have focused only on model discrimination or estimating a meaningful subset of the model parameters in Copula models (Perrone et al., 2016; Denman et al., 2011). However, our approach to discriminate between Copula models is much more flexible than what appears in the literature (Perrone et al., 2016). Indeed, the Ds-optimality approach of Perrone et al. (2016) could not be applied in Example 2 due to the consideration of many Copula models. The empirical findings in this study provide a new understanding of parameter estimation and model discrimination for Copula models in the experimental design context. These findings should be useful for experimenters when designing trials which observe more than a single univariate response. Overall, it appears relatively straightforward to estimate model parameters across the variety of Copula models considered in this work. However, there were particular challenges in terms of discrimination, and the results presented here should provide support in developing an appropriate set of rival Copula models.

The identified Bayesian sequential design approach can be generalised to different multivariate

outcomes and non-linear models. There are some situations where sequential design approaches are not applicable, such as experiments undertaken in batches. It may be possible to extend our sequential design approach to such settings through the use of efficient posterior approximations. Such approximations may include the Laplace approximation (Overstall et al., 2017). In studies where high-dimensional multivariate data are observed, it may be necessary to adopt particular Vine-Copulas (Brechmann et al., 2013; Panagiotelis et al., 2017). These are areas of research which we hope to explore in the future.

Acknowledgement

SGJS was supported by QUTPRA scholarship from the Queensland University of Technology. CCD was supported by an Australian Research Council's Discovery Early Career Researcher Award funding scheme (DE160100741). Computational resources and services used in this work were provided by the HPC and Research Support Group, Queensland University of Technology, Brisbane, Australia.

References

Borth, D. M. (1975), 'A total entropy criterion for the dual problem of model discrimination and parameter estimation', *Journal of the Royal Statistical Society. Series B (Methodological)* **37**(1), 77–87.

Brechmann, E. C., Schepsmeier, U. et al. (2013), 'Modeling dependence with c-and d-vine Copulas: The r-package cdvine', *Journal of Statistical Software* **52**(3), 1–27.

Carlin, B. P., Kadane, J. B. and Gelfand, A. E. (1998), 'Approaches for optimal sequential decision analysis in clinical trials', *Biometrics* **54**(3), 964–975.

Chaloner, K. and Verdinelli, I. (1995), 'Bayesian experimental design: A review', *Statistical Science* **10**(3), 273–304.

Clayton, D. G. (1978), 'A model for association in bivariate life tables and its application in epidemiological studies of familial tendency in chronic disease incidence', *Biometrika* **65**(1), 141–151.

Craiu, R. V. and Rosenthal, J. S. (2014), 'Bayesian computation via markov chain monte carlo', *Annual Review of Statistics and Its Application* **1**, 179–201.

Del Moral, P., Doucet, A. and Jasra, A. (2006), 'Sequential Monte Carlo samplers', *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **68**(3), 411–436.

Denman, N., McGree, J., Eccleston, J. and Duffull, S. B. (2011), 'Design of experiments for bivariate binary responses modelled by Copula functions', *Computational Statistics & Data Analysis* **55**(4), 1509–1520.

Drovandi, C. C., McGree, J. M. and Pettitt, A. N. (2013), 'Sequential Monte Carlo for Bayesian sequentially designed experiments for discrete data', *Computational Statistics & Data Analysis* **57**(1), 320–335.

Drovandi, C. C., McGree, J. M. and Pettitt, A. N. (2014), 'A sequential Monte Carlo algorithm to incorporate model uncertainty in Bayesian sequential design', *Journal of Computational and Graphical Statistics* **23**(1), 3–24.

Durante, F. and Sempi, C. (2010), Copula theory: an introduction, *in* 'Copula theory and its applications', Springer, pp. 3–31.

Frank, M. J. (1979), 'On the simultaneous associativity of $F(x, y)$ and $x + y - F(x, y)$', *Aequationes mathematicae* **19**(1), 194–226.

Genest, C. and MacKay, J. (1986), 'The joy of Copulas: bivariate distributions with uniform marginals', *The American Statistician* **40**(4), 280–283.

Gumbel, E. J. (1960), 'Distributions des valeurs extrêmes en plusieurs dimensions', *Publ. Inst. Statist. Univ. Paris* **9**, 171–173.

Joe, H. (1997), *Multivariate models and multivariate dependence concepts*, Chapman and Hall, London.

Kullback, S. and Leibler, R. A. (1951), 'On information and sufficiency', *The annals of mathematical statistics* **22**(1), 79–86.

McGree, J. M. (2017), 'Developments of the total entropy utility function for the dual purpose of model discrimination and parameter estimation in Bayesian design', *Computational Statistics and Data Analysis* **113**, 207–225.

McGree, J. M., Drovandi, C. C., White, G. and Pettitt, A. N. (2016), 'A pseudo-marginal sequential Monte Carlo algorithm for random effects models in Bayesian sequential design', *Statistics and Computing* **26**(5), 1121–1136.

Nelsen, R. (2006), 'An introduction to Copulas, 2nd', *New York: SpringerScience Business Media* .

Overstall, A. M., McGree, J. M. and Drovandi, C. C. (2017), 'An approach for finding fully Bayesian optimal designs using normal-based approximations to loss functions', *Statistics and Computing* pp. 1–16.
URL: *http://dx.doi.org/10.1007/s11222-017-9734-x*

Panagiotelis, A., Czado, C., Joe, H. and Stber, J. (2017), 'Model selection for discrete regular vine Copulas', *Computational Statistics & Data Analysis* **106**, 138–152.

Perrone, E. and Müller, W. G. (2016), 'Optimal designs for Copula models', *Statistics* **50**(4), 917–929.

Perrone, E., Rappold, A. and Müller, W. G. (2016), 'D_s-optimality in Copula models', *Statistical Methods & Applications* pp. 1–16.
URL: *http://dx.doi.org/10.1007/s10260-016-0375-6*

Schoelzel, C. and Friederichs, P. (2008), 'Multivariate non-normally distributed random variables in climate research–introduction to the Copula approach', *Nonlin. Processes Geophys.* **15**(5), 761–772.

Song, P. X.-K., Li, M. and Yuan, Y. (2009), 'Joint regression analysis of correlated data using gaussian Copulas', *Biometrics* **65**(1), 60–68.

Tao, A. (2010), A dose-finding method in joint modeling of efficacy and safety endpoints in phase II studies: An extension of the MCP-Mod method, PhD thesis, University of Medicine and Dentistry of New Jersey.

Tao, Y., Liu, J., Li, Z., Lin, J., Lu, T. and Yan, F. (2013), 'Dose-finding based on bivariate efficacy-toxicity outcome using archimedean Copula', *PloS one* **8**(11), e78805.

Tierney, L. (1994), 'Markov chains for exploring posterior distributions', *the Annals of Statistics* pp. 1701–1728.

Appendix

Example 1

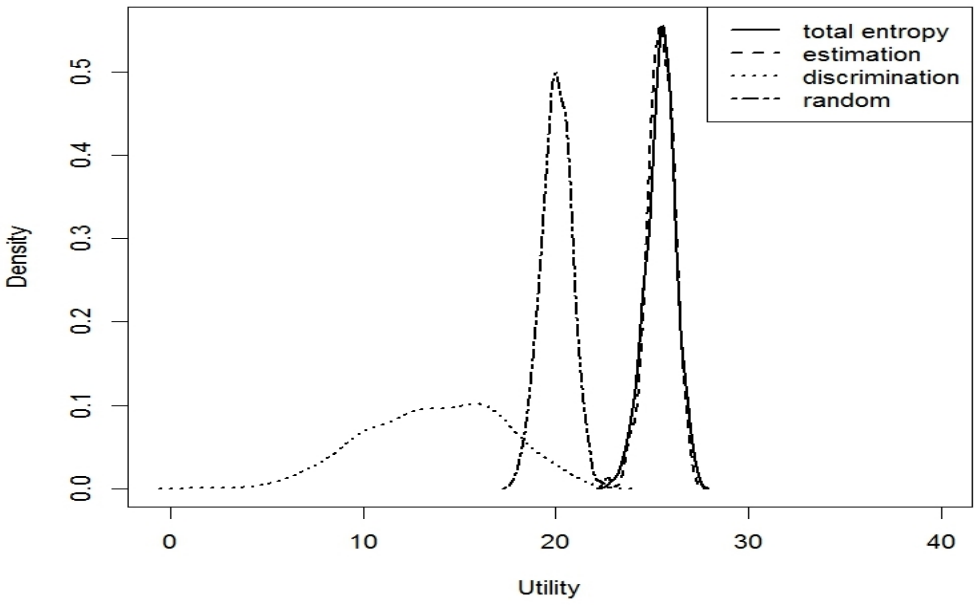

Figure 10: The distribution of the utilities (for estimation) at the final iteration of the SMC algorithm over the 500 simulations when Frank Copula is true.

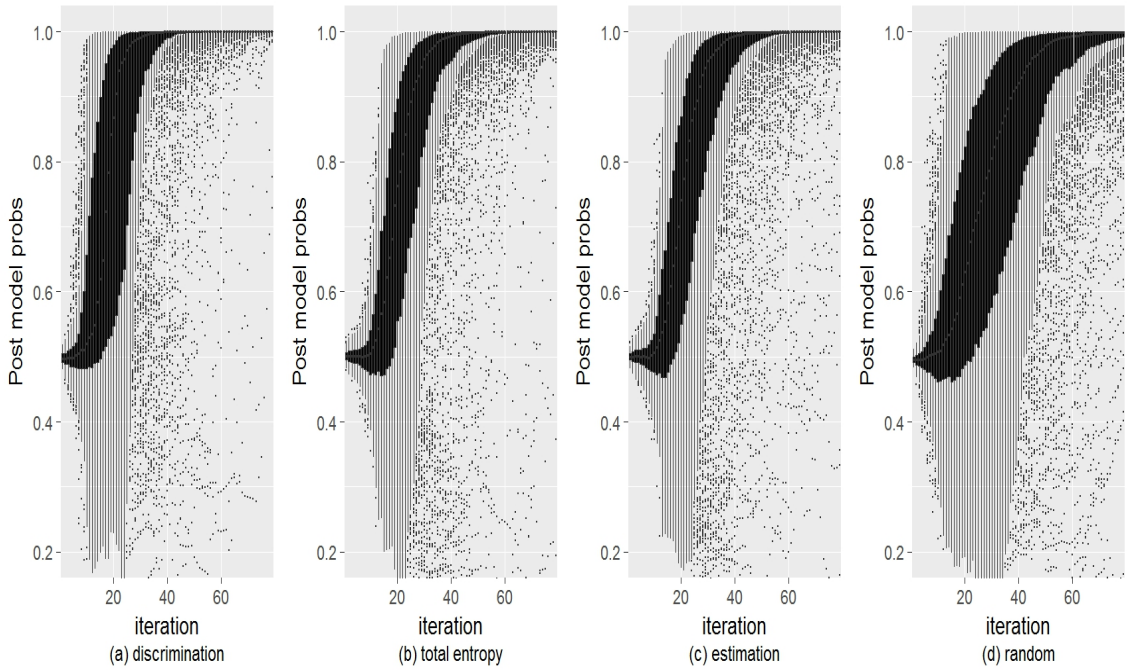

Figure 11: The distribution of the posterior model probabilities of the true Copula model over 500 runs for 250 subjects under each utility when Frank Copula is true.

Example 2

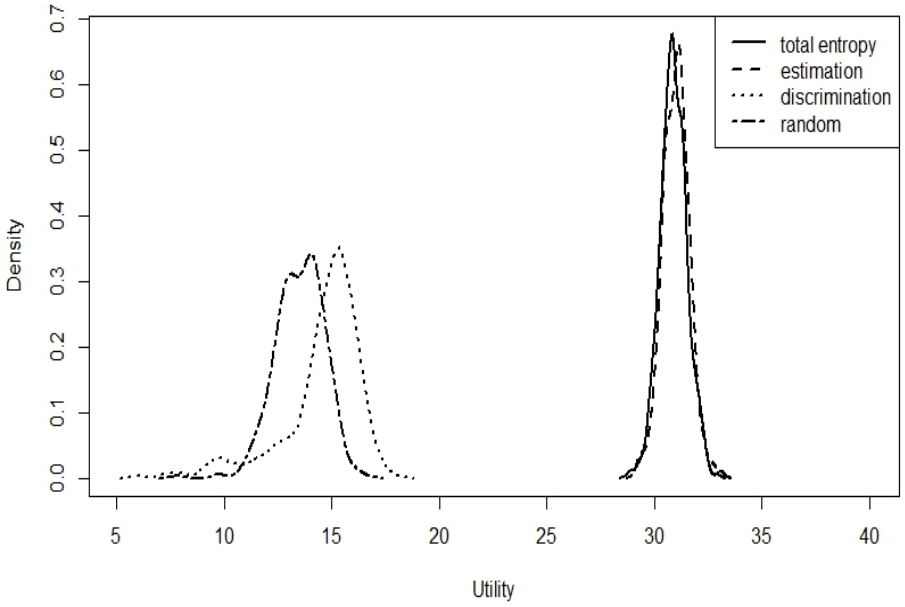

Figure 12: The distribution of the utilities (for estimation) at the final iteration of the SMC algorithm over the 500 simulations when Frank Copula is true.

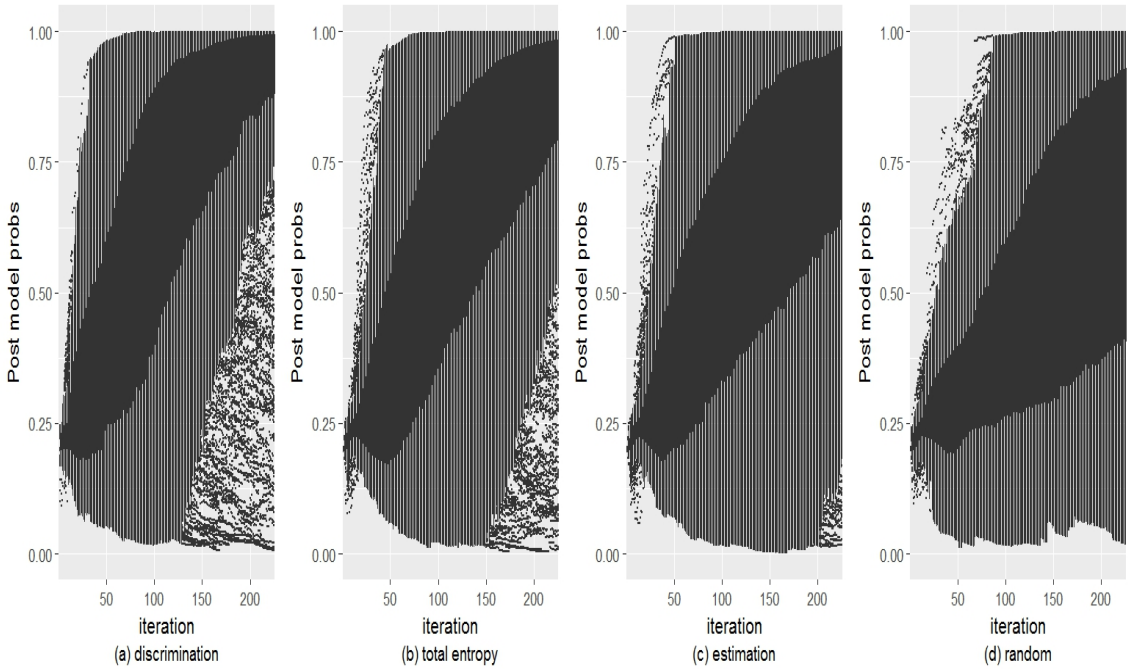

Figure 13: The distribution of the posterior model probabilities of the true Copula model over 500 runs for 225 subjects under each utility when Frank Copula is true.